A.F / IFW

PTO/SB/21 (01-08)
Approved for use through 06/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	10/642,203
Filing Date	August 18, 2003
First Named Inventor	Nobuyuki Enomoto
Art Unit	2619
Examiner Name	Lee, B.
Attorney Docket Number	MA-584-US
Total Number of Pages in This Submission	

ENCLOSURES *(Check all that apply)*

- [x] Fee Transmittal Form
  - [x] Fee Attached
- [ ] Amendment / Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Reply to Missing Parts/ Incomplete Application
  - [ ] Reply to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) ______
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [x] Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- [ ] Proprietary Information
- [ ] Status Letter
- [ ] Other Enclosure(s) (please identify below):

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	**McGinn Intellectual Property Law Group, PLLC** **8321 Old Courthouse Road, Suite 200, Vienna, Virginia 22182**		
Signature	Frederick Cooperrider		
Printed name	**Frederick E. Cooperrider, Esq.**		
Date	**September 30, 2008**	Reg. No.	**36,769**

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria,VA 22313-1450 on the date shown below:

Signature	**HAND DELIVERED**		
Typed or printed name		Date	

PTO/SB/21 (01-08)
Approved for use through 06/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission

Application Number	10/642,203
Filing Date	August 18, 2003
First Named Inventor	Nobuyuki Enomoto
Art Unit	2619
Examiner Name	Lee, B.
Attorney Docket Number	MA-584-US

ENCLOSURES *(Check all that apply)*

- ☒ Fee Transmittal Form
  - ☒ Fee Attached
- ☐ Amendment / Reply
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Reply to Missing Parts/ Incomplete Application
  - ☐ Reply to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) ______
  - ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☒ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☐ Other Enclosure(s) (please identify below):

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	**McGinn Intellectual Property Law Group, PLLC** **8321 Old Courthouse Road, Suite 200, Vienna, Virginia 22182**		
Signature	Frederick Cooperrider		
Printed name	**Frederick E. Cooperrider, Esq.**		
Date	**September 30, 2008**	Reg. No.	**36,769**

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria,VA 22313-1450 on the date shown below:

Signature	**HAND DELIVERED**		
Typed or printed name		Date	

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of

Enomoto, et al

Serial No.:	10/642,203	Group Art Unit:	2619
Filed:	August 18, 2003	Examiner:	Lee, B.

For: NETWORK SYSTEM, SPANNING TREE CONFIGURATION METHOD AND CONFIGURATION, PROGRAM, AND SPANNING TREE CONFIGURATION NODE

Commissioner of Patents
Alexandria, VA 22313-1450

APPELLANTS' BRIEF ON APPEAL

Sir:

Appellants respectfully appeal the rejection of claims 1, 2, 4, 7, 9, 10, 12-15, 17, 18, 20, 23, 25, 26, 28-31, 33, 35, 37, 38, 40, 41, 43, 46, 48, 50, 52, 53, 55, 56, and 58-61 in the Office Action mailed on April 9, 2008. A Notice of Appeal was timely filed on July 30, 2008.

I. REAL PARTY IN INTEREST

The real party in interest is NEC Corporation, assignee of 100% interest of the above-referenced patent application.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants, Appellants' legal representative or Assignee which would directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1, 2, 4-15, 17, 18, 20-31, 33, 35-46, 48, 50-61, and 63 are all the claims presently pending in the application. Claims 3, 16, 19, 32, 34, 47, 49, and 62 are canceled, and the Examiner indicates that claims 5, 6, 8, 11, 21, 22, 24, 27, 36, 39, 42, 51, 54, 57, and 63 would be allowable if rewritten in independent format and the indefiniteness rejection were overcome.

The Examiner objects to claims 1, 2, 4-15, 17, 18, 20-31, 33, 35-46, 48, 50-61, and 63 for various informalities. The Advisory Action mailed on July 14, 2008, did not indicate whether the claim amendments in the Amendment Under 37 CFR §1.116 filed on June 16, 2008, would be entered for purpose of Appeal, so these claim objections remain in the claims shown in the Claims Appendix, and the version of the claims in the Appendix is the version of the Amendment Under 37 CFR §1.111 filed on December 27, 2007.

Claims 1, 17, 33, and 48 stand rejected under 35 U.S.C. § 103(a) as unpatentable over U.S. Patent Publication No. 2003/0165119 to Hsu et al., further in view of JP patent 2002-353998 to Yasuaki and US Patent 5,761,435 to Fukada et al. Claims 4, 12-14, 20, 28-30, 35, 37, 38, 43-45, 50, 52, 53, and 58-62 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Yasuaki/Fukada, further in view of US Patent 6,882,630 to Seaman. Claims 9 and 10 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada/Seaman, further yet in view of US Patent 7,061,876 to Ambe. Claims 25, 26, 40, 41, 55, and 56 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada, further yet in view of Ambe.

All rejections are herein being appealed.

IV. STATUS OF AMENDMENTS

An Amendment Under 37 CFR §1.116 was filed on June 16, 2008. In the Advisory Action mailed on July 14, 2008, the Examiner indicated that the arguments in the Amendment Under 37 CFR §1.116 were not persuasive and that all rejections of record were maintained. Because the Examiner failed to indicate on the Advisory Action mailed on July 14, 2008, that this after-final amendment would be entered for purpose of Appeal, the claims of the Appendix are those claims as appearing in the Amendment Under 37 CFR§ 1.111, filed on December 27, 2007.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Conventional methods of setting up a spanning tree, as discussed beginning at line 13 on page 1 of the disclosure, affect a large area when a spanning tree has to be modified due to a failure.

In contrast, the present invention uses a self-configuring partial network to first configure itself and then configure the attached networks, if and as required, thereby shortening the configuration time.

This approach is completely different from the approach of autonomous rings, as taught in Hsu, used as the primary reference in the prior art rejections.

<u>The basis in the specification for the claims having rejections specifically argued by Appellants in § VII below is:</u>

1. (Rejected) A network system (see Figure 1) for setting a transfer path according to a spanning tree on a network connecting a plurality of nodes, wherein

two different networks (NETWORK2, NETWORK3, Figure 1; lines 11-20 of page 13) are connected by a self-configuring partial network (NETWORK1, Figure 1; lines 20-

22 of page 23 and line 18 of page 27 through line 8 of page 28) consisting of at least four nodes (NODEs 11-14, Figure 1) accommodating no terminal, and

a node (11, Figure 2) belonging to said self-configuring partial network configures (Configuration Unit100U, Figure 2) and manages (Tree Manager 104, 108, Figure 2) a spanning tree for every other network adjacent to the self-configuring partial network (line 15 of page 28 through line 19 of page 31), according to a spanning tree protocol (lines 6-10 of page 13),

wherein said node (11, Figure 2) comprises:

a plurality of transfer units (101, 105, Figure 2) which determines an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame (lines 15-20 of page 17); and

a plurality of tree managers (104, 108, Figure 2) which configures a spanning tree for every said partial network and said network, according to the spanning tree protocol and transfers a frame (lines 3-19 of page 18; lines 18-22 of page 19),

wherein said tree manager (104, Figure 4) comprises:

a tree controller (1041, Figure 4) which determines a state of a port according to the spanning tree protocol (lines 12-18 of page 21);

a BPDU transmitter/receiver (1042, 1043, 1044, Figure 4) which transmits and receives one or more control signals of the spanning tree protocol (line 19 of page 21 through line 11 of page 22); and

a port blocking unit (1045, 1046, 1047, Figure 4) which closes or opens a port (lines 3-4 of page 24).

4. (Rejected) The network system as set forth in claim 3, wherein said node (11, Figure 2) comprises:

said tree manager (104, Figure 2; lines 3-19 of page 18) which manages the spanning tree of the self-configuring partial network (Network 1, Figure 1); and

a virtual port (116, Figure 2; lines 6-23 of page 20) which packs into one of the output ports to said self-configuring partial network which connects to said transfer unit (105, Figure 2).

9. (Rejected) The network system as set forth in claim 4, wherein said node comprises:

an address learning unit (102,106, Figure 2; line 26 of page 18 through line 2 of page 19) which creates a table (103,107, Figure 2), based on an input port and a source MAC address of the received frame; and

a table (103,107, Figure 2) which determines an output destination port (1032, Figure 3; lines 3-7 of page 19) by using the destination MAC address (1031, Figure 3) as a key.

10. (Rejected) The network system as set forth in claim 9, wherein said table (103, Figure 3) comprises:

a destination MAC address field (1031, Figure 3) which describes the destination MAC address (line 27 of page 20 through line 2 of page 21); and

an output port field (1032, Figure 3) which describes an output destination port corresponding to the destination MAC address (lines 3-4 of page 21).

17. (Rejected) A node (11, Figure 2) that forms a spanning tree on a network connecting a plurality of nodes (Figure 1), the forming of the spanning tree comprising:

configuring a self-configuring partial network (NETWORK1, Figure 1; lines 20-22 of page 23 and line 18 of page 27 through line 8 of page 23) which connects two different networks (NETWORK2, NETWORK3, Figure 1; lines 11-20 of page 13) by, at least, four nodes (NODEs 11-14, Figure 1) accommodating no terminal (line 18 of page 27 through line 8 of page 28); and

configuring (Configuration Unit100U, Figure 2; line 9 of page 28 through line 14 of page 30) and managing a spanning tree for every other network adjacent to the self-

configuring partial network (line 15 of page 28 through line 19 of page 31), according to a spanning tree protocol (lines 6-10 of page 13),

wherein said node (11, Figure 2) comprises:

a plurality of transfer units (101, 105, Figure 2) which determines an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame (lines 15-20 of page 17); and

a plurality of tree managers (104, 108, Figure 2) which configures a spanning tree for every said partial network and said network, according to the spanning tree protocol and transfers a frame (lines 3-19 of page 18; lines 18-22 of page 19),

wherein said tree manager (104, Figure 4) comprises:

a tree controller (1041, Figure 4) which determines a state of a port according to the spanning tree protocol (lines 12-18 of page 21);

a BPDU transmitter/receiver (1042, 1043, 1044, Figure 4) which transmits and receives one or more control signals of the spanning tree protocol (line 19 of page 21 through line 11 of page 22); and

a port blocking unit (1045, 1046, 1047, Figure 4) which closes or opens a port (lines 3-4 of page 24).

20. (Rejected) The node as set forth in claim 19, said node (11, Figure 2) comprising

said tree manager (104, Figure 2) which manages the spanning tree of the self-partial network; and

a virtual port (116, Figure 2) which packs into one the output ports to said self-partial network which connects said transfer unit (105, Figure 2).

33. (Rejected) A spanning tree configuration method of configuring a spanning tree on a network connecting a plurality of nodes, comprising:

configuring a self-configuring partial network (NETWORK1, Figure 1; lines 20-22 of page 23 and line 18 of page 27 through line 8 of page 23) which connects two different

networks (NETWORK2, NETWORK3, Figure 1; lines 11-20 of page 13) by, at least, four nodes (NODEs 11-14, Figure 1) accommodating no terminal;

configuring (Configuration Unit100U, Figure 2) and managing a spanning tree for every other network adjacent to the self-configuring partial network (line 15 of page 28 through line 19 of page 31), according to a spanning tree protocol (lines 6-10 of page 13);

a transfer step of determining an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame (line 15 of page 17 through line 2 of page 18); and

a tree manager step of configuring a spanning tree for every said self-configuring partial network, according to the spanning tree protocol and transferring a frame,

wherein said tree manager step comprises:

a tree controller step of determining a state of a port according to the spanning tree protocol (lines 5-12 of page 18);

a BPDU transmitting/receiving step (1042, 1043, 1044, Figure 4) of transmitting and receiving one or more control signals of the spanning tree protocol (lines 13-16 of page 18; line 19 of page 21 through line 11 of page 22); and

a port blocking step (1045, 1046, 1047, Figure 4) of closing or opening a port (lines 16-19 of page 18; lines 3-4 of page 24).

48. (Rejected) A spanning tree configuration program comprising a sequence of machine-readable instructions encoded on a machine-readable medium for running on each node forming a spanning tree on a network connecting a plurality of nodes, said sequence of instructions comprising the following functions of:

configuring a self-configuring partial network (NETWORK1, Figure 1; lines 20-22 of page 23 and line 18 of page 27 through line 8 of page 23) which connects two different networks (NETWORK2, NETWORK3, Figure 1; lines 11-20 of page 13) said self-configuring partial network comprising, at least, four nodes (NODEs 11-14, Figure 1) accommodating no terminal;

configuring (Configuration Unit100U, Figure 2) and managing a spanning tree for every other network adjacent to the self-configuring partial network (line 15 of page 28 through line 19 of page 31), according to a spanning tree protocol (lines 6-10 of page 13);

a transfer function of determining an output destination port in every said partial network, based on a destination MAC address of an input frame; and

a tree manager function of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame;

wherein said tree manager function comprises:

a tree controller function of determining a state of a port according to the spanning tree protocol;

a BPDU transmitting/receiving function (1042, 1043, 1044, Figure 4) of transmitting and receiving one or more control signals of the spanning tree protocol (line 19 of page 21 through line 11 of page 22); and

a port blocking function (1045, 1046, 1047, Figure 4) of closing or opening a port (lines 3-4 of page 24).

50. (Rejected) The spanning tree configuration program as set forth in claim 49, comprising

a function of connecting said tree manager function (104, Figure 2) of managing the spanning tree of the self-configuring partial network and said transfer function (105, Figure 2) through a virtual port (116, Figure 2) for putting the output port toward the self-configuring partial network (Network 1, Figure 1) together.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Appellants present the following grounds for review by the Board of Patent Appeals and Interferences:

GROUND 1: Claims 1, 17, 33, and 48 stand rejected under 35 U.S.C. § 103(a) as unpatentable over U.S. Patent Publication No. 2003/0165119 to Hsu et al., further in view of JP patent 2002-353998 to Yasuaki and US Patent 5,761,435 to Fukada et al;

GROUND 2: Claims 4, 12-14, 20, 28-30, 35, 37, 38, 43-45, 50, 52, 53, and 58-62 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Yasuaki/Fukada, further in view of US Patent 6,882,630 to Seaman; and

GROUND 3: Claims 9 and 10 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada/Seaman, further yet in view of US Patent 7,061,876 to Ambe, and claims 25, 26, 40, 41, 55, and 56 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada, further yet in view of Ambe.

VII. ARGUMENTS

In general, Appellants respectfully submit that primary reference Hsu clearly teaches against the claimed invention, so that all reliance upon this reference depends upon totally ignoring that its method fundamentally contradicts the method of the claimed invention.

Moreover, Appellants submit that all references relied upon as secondary, tertiary, and higher order references are based upon the spanning tree protocol, which primary reference Hsu attempts to replace, as the essence of its method. Therefore, it would be improper to use these references in combination with primary reference Hsu, since the methods and details described in these references would defeat the purpose of Hsu to provide an alternative to the configuring a spanning tree in accordance with a spanning tree protocol, as described in MPEP §2143.01: "*If proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. In re Gordon, 733 F.2d 900, 221USPQ 1125 (Fed. Cir. 1984).*"

Additionally, since the primary reference Hsu attempts to replace the spanning tree protocol by a method based on rings, these supporting references that inherently rely upon the spanning tree protocol would clearly change the principle of operation of primary reference Hsu. Therefore, modification of Hsu by a reference that changes its principle of operation would be improper, and the rejection of record would clearly fail to establish a *prima facie* rejection under the holding of *In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959), as described in MPEP § 2143.01: "*If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie obvious.*"

Finally, Appellants respectfully submit that the rejections of record fail to provide a reasonable objective rationale to modify primary reference Hsu, since all rejections are based on rationales that are merely conclusory statements that ignore the teachings of primary reference Hsu.

GROUND 1: The Obviousness Rejection for claims 1, 17, 33, and 48 Based on U.S. Patent Publication No. 2003/0165119 to Hsu et al., further in view of JP patent 2002-353998 to Yasuaki and US Patent 5,761,435 to Fukada et al.

The Examiner's Position

The Examiner alleges that "... *Hsu teaches two different networks are connected by nodes, the node belonging to self-configuring partial networks configures and manages a ring (see Fig. 15). Hsu does teach that the invention may also be used with spanning tree protocol networks (see paragraph 56 lines 3-8) and a port blocking unit which closes or opens a port (see paragraph 59 lines 14-18). Hsu teaches all the subject matter of the claimed invention with the exception of two different networks are connected by a self-configuring partial network consisting of at least four nodes, the node configuring and managing a spanning tree network, determining an output destination port from a destination MAC adddress and tree managers, and a tree controller and BPDU transmitter/receiver.*" (Page 4 of Office Action mailed on April 9, 2008)

The Examiner also alleges that secondary reference Yasuaki "... *teaches two different networks (see Fig. 9 items 42 and 43) are mutually connected by a partial network (41) comprising at leat four nodes (bridges) accommodating no terminal, and managing the network, where the nodes belonging to the partial network create a spanning tree for [each] network adjacent to its own partial network according to the spanning tree protocol (see Fig. 9 Items 20 and 30).*" (see Page 4 of Office Action)

The Examiner considers that "... *it would have been obvious to one of ordinary skill in the art to create a dedicated ring as taught by Yasuaki to manage the two different networks of Hsu to decrease the time required to restore service after an interruption.*"

Finally, the Examiner alleges that Hsu, as modified by Yasuaki, "... *teaches all the subject matter of the claimed invention with the exception of determining an output destination port from a destination MAC address and tree managers, and a tree controller and BPDU transmitter/receiver*" and relies upon tertiary reference Fukuda to overcome these deficiencies. (Last sentence of page 4 of the Office Action)

According to the Examiner, the motivation to modify Hsu/Yasuaki with Fukuda is "… *to make the system more flexible*." (Page 5 of the Office Action)

Appellants' Position

Appellants respectfully submits that the rejection summarized above fails to meet the initial burden of a *prima facie* obviousness rejection, since the Examiner mischaracterizes these references by taking words out of context, thereby clearly engaging in improper hindsight based on using the claimed invention as a roadmap.

Moreover, Appellants submit that the combination of these three references would not result in the invention defined by these four independent claims.

More specifically, Appellants begin by pointing out that, to one having ordinary skill in the art, primary reference Hsu can only be considered as actually teaching against the approach of the claimed invention, since it provides an alternate approach to configuring a network into a spanning tree using a central location.

That is, as clearly explained in even the Abstract alone, primary reference Hsu teaches that a large network having many bridges is to be configured by considering the large network to be built as a combination of smaller networks, many of which will be arranged in a ring topology. Each ring topology will configure itself by having one bridge that serves as a master bridge having the role to block data to preclude a loop within that ring.

Appellants submit that this approach of Hsu, of forming autonomously-configuring rings within a larger network, differs fundamentally from that of the claimed invention by at least the following:

- Hsu clearly teaches the disadvantage of the spanning tree algorithm (paragraph [00006]) and even describes its technique as an alternative to the spanning tree protocol (paragraph [0007]).
- Hsu's technique, therefore, inherently teaches against using one ring to configure another ring in the network. The closest that such configuration of another ring occurs in Hsu's technique is that two adjacent rings having a

> shared link that fails will cause the two rings to merge to form a larger single ring (see paragraphs [0010] through [0012]) having, again, a single master bridge for precluding a loop within the merged, larger ring.

Thus, contrary to the Examiner's characterization of primary reference Hsu, Appellants respectfully submit that, to one having ordinary skill in the art, Hsu clearly teaches an approach that directly contradicts the requirements of the description of the four independent claims, since the rings of Hsu, even if considered as self-configuring subnetworks, do not configure an adjacent ring, let alone configure and manage a spanning tree in that adjacent network. Moreover, the self-configuration of the ring in Hsu merely involves the selection of a master bridge, rather than the calculation of a spanning tree.

As noted above, in paragraph [0006] of Hsu, this primary reference clearly teaches against using the spanning tree algorithm and can only, therefore, be described as clearly teaching against the very modification needed to Hsu necessary to satisfy the four rejected independent claims.

Hence, turning to the clear language of the claims, in Hsu there is no teaching or suggestion of: "... a node belonging to said self-configuring partial network configures and manages a spanning tree for every other network adjacent to the self-configuring partial network, according to a spanning tree protocol ....", as required by independent claim 1, and independent claims 17, 33, and 48 have similar language.

Relative to the effect of secondary reference Yasuaki, Appellants submit that, as best understood from the Abstract and the English translation provided by the Examiner in the Advisory Action mailed on July 14, 2008, this reference clearly describes a spanning tree method having a novelty that the spanning tree information, once the networks are configured into spanning trees, is stored in a communication device so that the spanning trees can be reconfigured more quickly upon a failure.

Appellants submit that this spanning tree approach for networks is directly contradictory to the teaching of primary reference Hsu, wherein is taught autonomous self-configuring rings that not only inherently operate on a different principle of operation for

the initial configuration of the network (*e.g.*, not using the spanning tree algorithm) but also operate on a different principle of operation upon a failure.

Therefore, Appellants submit that, in view of the contradictory teachings of primary reference Hsu, one having ordinary skill in the art would not have been motivated to use any teaching in secondary reference JP 2002-353998 to Yasuaki to modify the method of Hsu.

Moreover, since secondary reference Yasuaki fails to suggest using a self-configuring network to configure another adjacent network, this reference does not overcome the deficiency identified above for primary reference Hsu. Thereby, even if Yasuaki were to be used to modify Hsu, the combination would not result in the claimed invention.

In the Advisory Action mailed on July 14, 2008, the Examiner alleges that Yasuaki teaches a self-configuring partial network (e.g., ring 41 in Figure 9) and that this ring 41 is configures networks to which it is adjacently attached, pointing to paragraphs 45 and 47 of the English translation.

In response, Appellants respectfully submit that, even if ring 41 of Yasuaki were to be considered as self configuring (e.g., similar to the method described in Hsu), there is no suggestion in paragraphs 45 and 47 that ring 41 thereinafter configures either of its adjacent rings 42,43, let along configuring by using a spanning tree algorithm.

Relative to the rationale in the rejection of record to modify primary reference Hsu by secondary reference Yasuaki (e.g., "... *to decrease the time required to restore service after an interruption*"), Appellants submit that primary reference Hsu already has such restoration capability (*e.g.*, merging two rings) and its restoration method would clearly be faster that the reconstruction method of Yasuaki based on a spanning tree restoration. It is noted that Yasuaki's claim for faster restoration service clearly refers to its prior art discussion, since the method of Yasuaki involves storing the spanning tree information in a table, thereby allowing this table information to provide a faster restoration than completely recalculating the spanning trees all over again after a failure. Appellants

submit that there is no credible reason of record to believe that such restoration to restore a spanning tree based on a table would be faster than the ring-merging method of Hsu.

Tertiary reference Fukuda is not used in any manner that would overcome the above-identified deficiency of primary reference Hsu. However, since Fukuda is also clearly reliant upon a spanning tree protocol method, Appellants submit that one having ordinary skill in the art would not have any reason to modify primary reference Hsu by Fukuda, absent impermissible hindsight. Moreover, even if such combination were to be made, the fundamental deficiency of Hsu would not be overcome.

Relative to the rationale to modify Hsu by tertiary reference Fukuda, wherein the Examiner alleges that one of ordinary skill in the art would have been motivated to modify Hsu in order "*... to make the system more flexible*", Appellants respectfully submit that Hsu is already "flexible" and that adding a spanning tree method into Hsu would only defeat its purpose, thereby being improper.

Moreover, there is no credible evidence of record that the method of Fukuda would somehow make the method of Hsu, clearly operating on another principle of operation, any more flexible.

For the reasons stated above, the claimed invention is fully patentable over the cited references, and the Board is respectfully requested to reverse these rejections for claims 1, 17, 33, and 48, based on Hsu/Yasuaki/Fukada.

GROUND 2: Claims 4, 12-14, 20, 28-30, 35, 37, 38, 43-45, 50, 52, 53, and 58-62 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Yasuaki/Fukada, further in view of US Patent 6,882,630 to Seaman

The Examiner's Position

Relative to claims 4, 20, and 50, the Examiner concedes that Hsu fails to utilize a virtual port and alleges that quadriary reference Seaman teaches a virtual port (e.g., 109 of Figure 1) and that one of skill in the art would have been motivated to modify Hsu to "... *use the system of Fukuda in the system of Hsu ... to make the system more flexible*."

Appellants' Position

Appellants begin by pointing out that the virtual port (e.g., 116 of Figure 2) of the claimed invention is an internal port clearly distinct from the normal ports 111-114 that connect to the two adjacent networks. That is, as shown in Figure 2, the virtual port 116 is interconnected between one tree manager 104 and the transfer unit 105, and, as explained in lines 6-8 of page 20 of the specification, relays transmission and reception of the frame between the tree manager 104 and the transfer unit 105. As further explained at lines 15-21 of page 30, this virtual port allows the tree manager 104 for network 2 to be able to recognize the ports 113 and 114 of network 1.

Contrary to the Examiner's characterization, there is no corresponding internal port in Seaman. Label 109 refers to the LLC (link control layer) that is the logical entity associated with the normal port 101 and does not reasonably correspond to an internal port that is described as a "virtual port" in the present application. That is, LLC 109 is merely a component of the normal port in Seaman. Moreover, primary reference Hsu is not demonstrated as missing a logical entity for its normal ports, and there is no reasonable rationale to add a virtual port to primary reference Hsu, let alone a virtual port as described in the present invention, since there is no credible reason to believe that the ports of Hsu would be any more efficient if a virtual port were to be added or that a virtual port would in any reasonable benefit Hsu.

For the reasons stated above, the claimed invention is fully patentable over these cited references, and the Board is respectfully requested to reverse these rejections.

GROUND 3: Claims 9 and 10 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada/Seaman, further yet in view of US Patent 7,061,876 to Ambe, and claims 25, 26, 40, 41, 55, and 56 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hsu/Fukada, further yet in view of Ambe.

The Examiner's Position

The Examiner concedes that Hsu fails to incorporate a address learning unit or a table but alleges that Ambe demonstrates these components and that one having skill in the art would have been motivated to modify Hsu "... *to make the system more efficient by building a routing table*."

Appellants' Position

Appellants respectfully submit that the rejection for these two claims fails to establish a *prima facie* obviousness rejection. There is no demonstration that primary reference Hsu would in any way benefit from these missing components. Moreover, the rationale of record is merely a conclusory allegation by the Examiner that Hsu would somehow be more efficient with such missing element. However, the Examiner fails to provide any credible support for such allegation.

For the reasons stated above, the claimed invention is fully patentable over the cited references, and the Board is respectfully requested to reverse these rejections.

IX. CONCLUSION

In view of the foregoing, Appellants submit that claims 1, 2, 4-15, 17, 18, 20-31, 33, 35-46, 48, 50-61, and 63, all the claims presently pending in the application, are clearly enabled and patentably distinct from the prior art of record and in condition for allowance. Thus, the Board is respectfully requested to remove all rejections of claims 1, 4, 9, 10, 12-14, 17, 20, 25, 26, 28-30, 33, 35, 37, 38, 40, 41, 43-45, 48, 50, 52, 53, 55, 56, and 58-62

Please charge any deficiencies and/or credit any overpayments necessary to enter this paper to Attorney's Deposit Account number 50-0481.

Respectfully submitted,

Dated: 09/30/08

Frederick E. Cooperrider
Reg. No. 36,769

McGinn Property Law Group, PLLC
8231 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817
(703) 761-4100
Customer Number: 21254

CLAIMS APPENDIX

The claims, as reflected upon entry of the Amendment Under 37 CFR §1.111 filed on December 27, 2007, are shown below:

1. (Rejected) A network system for setting a transfer path according to a spanning tree on a network connecting a plurality of nodes, wherein

two different networks are connected by a self-configuring partial network consisting of at least four nodes accommodating no terminal, and

a node belonging to said self-configuring partial network configures and manages a spanning tree for every other network adjacent to the self-configuring partial network, according to a spanning tree protocol,

wherein said node comprises:

a plurality of transfer units which determines an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame; and

a plurality of tree managers which configures a spanning tree for every said partial network and said network, according to the spanning tree protocol and transfers a frame,

wherein said tree manager comprises:

a tree controller which determines a state of a port according to the spanning tree protocol;

a BPDU transmitter/receiver which transmits and receives one or more control signals of the spanning tree protocol; and

a port blocking unit which closes or opens a port.

2. (Rejected) The network system as set forth in claim 1, where

said self-configuring partial network is formed by a link connecting opposite nodes, and

each pair of the nodes for the same number as forming said self-configuring partial network is connected to each different network.

3. (Canceled)

4. (Rejected) The network system as set forth in claim 3, wherein said node comprises:

said tree manager which manages the spanning tree of the self-configuring partial network; and

a virtual port which packs into one of the output ports to said self-configuring partial network which connects to said transfer unit.

5. (Allowable) The network system as set forth in claim 4, where said node comprises:

transfer units which determine an output destination port in every said partial network, based on a destination MAC address of an input frame;

an RPR frame transfer unit which determines a destination RPR address, a ring ID, and an output destination port, based on the destination MAC address of the input frame;

tree managers which configure a spanning tree for every said partial network, according to the spanning tree protocol and transfer a frame;

a TTL manager which performs subtraction of TTL and discards the frame by the TTL; and

the virtual port which connects said tree manager for managing the spanning tree of the self-configuring partial network and said RPR frame transfer unit and puts the output port toward the self-configuring partial network together.

6. (Allowable) The network system as set forth in claim 5, wherein said TTL manager comprises:

a TTL checker which discards the frame with reference to a TTL value; and

a TTL controller which performs addition and subtraction of the TTL value.

7. (Rejected) The network system as set forth in claim 1, wherein said node comprises:

a plurality of transfer units which determines an output destination port in every partial network, based on a destination MAC address of an input frame,

a plurality of tree managers which configures a spanning tree for every said partial network, according to the spanning tree protocol and transfers a frame, and

a BPDU identifying unit which determines a tree manager of an output destination of an input BPDU frame according to an identifier.

8. (Allowable) The network system as set forth in claim 7, wherein said BPDU identifying unit comprises:

an identifier inserting unit which inserts a tag or a bit (tags or bits) for identifying the tree manager; and

an identifier deleting unit which deletes the tag or the bit (tags or bits) used for identifying the tree manager.

9. (Rejected) The network system as set forth in claim 4, wherein said node comprises:

an address learning unit which creates a table, based on an input port and a source MAC address of the received frame; and

a table which determines an output destination port by using the destination MAC address as a key.

10. (Rejected) The network system as set forth in claim 9, wherein said table comprises:

a destination MAC address field which describes the destination MAC address; and

an output port field which describes an output destination port corresponding to the destination MAC address.

11. (Allowable) The network system as set forth in claim 1, wherein said node comprises:

a plurality of transfer units which determines an output destination port in every said partial network, based on an identification tag of an input frame;

a multiphase tree manager which configures a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a virtual port which connects said multiphase tree manager and said transfer unit and puts the output port toward the self-configuring partial network together.

12. (Rejected) The network system as set forth in claim 3, wherein said node comprises a failure detector which detects a failure through transmission and receipt of keep alive frames.

13. (Rejected) The network system as set forth in claim 12, wherein said failure detector comprises:

a signal separator which separates the keep alive frames from the other frame; and

a keep alive signal transmitter/receiver which transmits and receives the keep alive frames.

14. (Rejected) The network system as set forth in claim 12, wherein said node comprises

a frame blocking unit which cuts off the port at a time of double failure.

15. (Rejected) The network system as set forth in claim 1, wherein said node comprises:

a plurality of transfer units which determines an output destination port in every said partial network, based on an identification tag of the input frame;

a multiphase tree manager which configures a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a tag operation unit which inserts and deletes an identification tag.

16. (Canceled)

17. (Rejected) A node that forms a spanning tree on a network connecting a plurality of nodes, the forming of the spanning tree comprising:

configuring a self-configuring partial network which connects two different networks by, at least, four nodes accommodating no terminal; and

configuring and managing a spanning tree for every other network adjacent to the self-configuring partial network, according to a spanning tree protocol,

wherein said node comprises:

a plurality of transfer units which determines an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame; and

a plurality of tree managers which configures a spanning tree for every said partial network and said network, according to the spanning tree protocol and transfers a frame,

wherein said tree manager comprises:

a tree controller which determines a state of a port according to the spanning tree protocol;

a BPDU transmitter/receiver which transmits and receives one or more control signals of the spanning tree protocol; and

a port blocking unit which closes or opens a port.

18. (Rejected) The node as set forth in claim 17, wherein said configuring the self-configuring partial network comprises:

configuring said self-configuring partial network by a link connecting said opposite nodes; and

connecting each pair of the nodes for the same number as forming said self-configuring partial network to each different network.

19. (Canceled)

20. (Rejected) The node as set forth in claim 19, said node comprising

said tree manager which manages the spanning tree of the self-partial network; and

a virtual port which packs into one the output ports to said self-partial network which connects said transfer unit.

21. (Allowable) The node as set forth in claim 19, said node comprising:

the several transfer units which determines an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame;

an RPR frame transfer unit which determines a destination RPR address, a ring ID, and an output destination port, based on the destination MAC address of the input frame;

the several tree managers which configures a spanning tree for said self-configuring partial network, according to the spanning tree protocol and transfers a frame;

a TTL manager which performs subtraction of TTL and discards the frame by the TTL; and

the virtual port for connecting said tree manager which manages the spanning tree of the self-configuring partial network and said RPR frame transfer unit and putting the output port toward the self-configuring partial network together.

22. (Allowable) The node as set forth in claim 21, wherein said TTL manager comprises:

a TTL checker which discards the frame with reference to a TTL value; and

a TTL controller which performs addition and subtraction of the TTL value.

23. (Rejected) The node as set forth in claim 18, comprising:

a plurality of transfer units which determines an output destination port in every said partial network, based on a destination MAC address of the input frame;

a plurality of tree managers which configures a spanning tree for every said partial network, according to the spanning tree protocol and transfers a frame; and

a BPDU identifying unit which determines a tree manager of an output destination of an input BPDU frame according to an identifier.

24. (Allowable) The node as set forth in claim 23, wherein said BPDU identifying unit comprises:

an identifier inserting unit which inserts a tag or a bit (tags or bits) for identifying the tree manager; and

an identifier deleting unit which deletes the tag or the bit (tags or bits) used for identifying the tree manager.

25. (Rejected) The node as set forth in claim 19, comprising:

an address learning unit which creates a table, based on an input port and a source MAC address of the received frame; and

a table which determines an output destination port by using the destination MAC address as a key.

26. (Rejected) The node as set forth in claim 25, wherein said table comprises:

a destination MAC address field which describes the destination MAC address; and

an output port field which describes an output destination port corresponding to the destination MAC address.

27. (Allowable) The node as set forth in claim 18, comprising:

a plurality of transfer units which determines an output destination port in every said partial network, based on an identification tag of an input frame;

a multiphase tree manager which configures a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a virtual port which connects said multiphase tree manager and said transfer unit and puts the output port toward the self-configuring partial network together.

28. (Rejected) The node as set forth in claim 19, comprising a failure detector which detects a failure through transmission and receipt of keep alive frames.

29. (Rejected) The node as set forth in claim 28, wherein said failure detector comprises:

a signal separator which separates the keep alive frames from the other frame; and

a keep alive signal transmitter/receiver which transmits and receives the keep alive frames.

30. (Rejected) The node as set forth in claim 28, comprising a frame blocking unit which cuts off the port at a time of double failure.

31. (Rejected) The node as set forth in claim 18, comprising:

a plurality of transfer units which determines an output destination port in every said partial network, based on an identification tag of the input frame;

a multiphase tree manager which configures a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a tag operation unit which inserts and deletes an identification tag.

32. (Canceled)

33. (Rejected) A spanning tree configuration method of configuring a spanning tree on a network connecting a plurality of nodes, comprising:

configuring a self-configuring partial network which connects two different networks by, at least, four nodes accommodating no terminal;

configuring and managing a spanning tree for every other network adjacent to the self-configuring partial network, according to a spanning tree protocol;

a transfer step of determining an output destination port in every said self-configuring partial network, based on a destination MAC address of an input frame; and

a tree manager step of configuring a spanning tree for every said self-configuring partial network, according to the spanning tree protocol and transferring a frame,

wherein said tree manager step comprises:

a tree controller step of determining a state of a port according to the spanning tree protocol;

a BPDU transmitting/receiving step of transmitting and receiving one or more control signals of the spanning tree protocol; and

a port blocking step of closing or opening a port.

34. (Canceled)

35. (Rejected) The spanning tree configuration method as set forth in claim 34, comprising:

said tree manager step managing the spanning tree of the self-configuring partial network,

connecting a virtual port which packs into one the output ports to said self-configuring partial network.

36. (Allowable) The spanning tree configuration method as set forth in claim 34, comprising:

said transfer step of determining an output destination port in every said partial network, based on a destination MAC address of an input frame;

an RPR frame transfer step of determining a destination RPR address, a ring ID, and an output destination port, based on the destination MAC address of the input frame;

said tree manager step of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame;

a TTL manager step of performing subtraction of TTL and discarding the frame by the TTL; and

a step of connecting said tree manager step of managing the spanning tree of the self-partial network and said RPR frame transfer step through a virtual port for putting the output port toward the self-partial network together.

37. (Rejected) The spanning tree configuration method as set forth in claim 34, wherein said TTL manager step comprises:

a TTL checker step of discarding the frame with reference to a TTL value; and

a TTL controller step of performing addition and subtraction of the TTL value.

38. (Rejected) The spanning tree configuration method as set forth in claim 35, comprising:

said transfer step of determining an output destination port in every said partial network, based on a destination MAC address of the input frame;

said tree manager step of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame; and

a BPDU identifying step of determining a tree manager step of an output destination of an input BPDU frame according to an identifier.

39. (Allowable) The spanning tree configuration method as set forth in claim 38, wherein said BPDU identifying step comprises:

an identifier inserting step of inserting a tag or a bit (tags or bits) for identifying said tree manager step; and

an identifier deleting step of deleting the tag or the bit (tags or bits) used for identifying said tree manager step.

40. (Rejected) The spanning tree configuration method as set forth in claim 34, comprising:

an address learning step of creating a table for determining an output destination port by using the destination MAC address as a key, based on an input port and a source MAC address of the received frame.

41. (Rejected) The spanning tree configuration method as set forth in claim 40, wherein said table comprises:

a destination MAC address field which describes the destination MAC address; and

an output port field which describes an output destination port corresponding to the destination MAC address.

42. (Allowable) The spanning tree configuration method as set forth in claim 33, comprising:

a transfer step of determining an output destination port in every said partial network, based on an identification tag of an input frame;

a multiphase tree manager step of configuring a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a step of connecting said multiphase tree manager step and said transfer step through a virtual port for putting the output port toward the self-configuring partial network together.

43. (Rejected) The spanning tree configuration method as set forth in claim 33, comprising a failure detecting step of detecting a failure through transmission and receipt of keep alive frames.

44. (Rejected) The spanning tree configuration method as set forth in claim 43, wherein said failure detecting step comprises:

a signal separating step of separating the keep alive frames from the other frame; and

a keep alive signal transmitting/receiving step of transmitting and receiving the keep alive frames.

45. (Original) The spanning tree configuration method as set forth in claim 43, comprising a blocking step of cutting off the port at a time of double failure.

46. (Rejected) The spanning tree configuration method as set forth in claim 33, comprising:

a transfer step of determining an output destination port in every said partial network, based on an identification tag of the input frame;

a multiphase tree manager step of configuring a spanning tree for every partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a tag operating step of inserting and deleting an identification tag.

47. (Rejected) The spanning tree configuration method as set forth in claim 34, wherein said multiphase tree manager step comprises:

a tree controller step of determining a state of a port according to the spanning tree protocol;

a BPDU transmitting/receiving step of transmitting and receiving a control signal (control signals) of the spanning tree protocol; and

a port blocking step of closing or opening a port.

48. (Rejected) A spanning tree configuration program comprising a sequence of machine-readable instructions encoded on a machine-readable medium for running on each node forming a spanning tree on a network connecting a plurality of nodes, said sequence of instructions comprising the following functions of:

configuring a self-configuring partial network which connects two different networks said self-configuring partial network comprising, at least, four nodes accommodating no terminal;

configuring and managing a spanning tree for every other network adjacent to the self-configuring partial network, according to a spanning tree protocol;

a transfer function of determining an output destination port in every said partial network, based on a destination MAC address of an input frame; and

a tree manager function of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame;

wherein said tree manager function comprises:

a tree controller function of determining a state of a port according to the spanning tree protocol;

a BPDU transmitting/receiving function of transmitting and receiving one or more control signals of the spanning tree protocol; and

a port blocking function of closing or opening a port.

49. (Canceled)

50. (Rejected) The spanning tree configuration program as set forth in claim 49, comprising

a function of connecting said tree manager function of managing the spanning tree of the self-configuring partial network and said transfer function through a virtual port for putting the output port toward the self-configuring partial network together.

51. (Allowable) The spanning tree configuration program as set forth in claim 48, comprising:

the transfer function of determining an output destination port in every said partial network, based on a destination MAC address of an input frame;

an RPR frame transfer function of determining a destination RPR address, a ring ID, and an output destination port, based on the destination MAC address of the input frame;

the tree manager function of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame;

a TTL manager function of performing subtraction of TTL and discarding the frame by the TTL; and

a function of connecting said tree manager function of managing the spanning tree of the self-configuring partial network and said RPR frame transfer function through a virtual port for putting the output port toward the self-configuring partial network together.

52. (Rejected) The spanning tree configuration program as set forth in claim 49, wherein said TTL manager function comprises a TTL checker function of discarding the frame with reference to a TTL value, and a TTL controller function of performing addition and subtraction of the TTL value.

53. (Rejected) The spanning tree configuration program as set forth in claim 50, comprising:

the transfer function of determining an output destination port in every said partial network, based on a destination MAC address of the input frame;

the tree manager function of configuring a spanning tree for every said partial network, according to the spanning tree protocol and transferring a frame; and

a BPDU identifying function of determining a tree manager function of an output destination of an input BPDU frame according to an identifier.

54. (Allowable) The spanning tree configuration program as set forth in claim 53, wherein said BPDU identifying function comprises:

an identifier inserting function of inserting a tag or a bit (tags or bits) for identifying said tree manager function; and

an identifier deleting function of deleting the tag or the bit (tags or bits) used for identifying said tree manager function.

55. (Rejected) The spanning tree configuration program as set forth in claim 49, comprising an address learning function of creating a table for determining an output destination port by using the destination MAC address as a key, based on an input port and a source MAC address of the received frame.

56. (Rejected) The spanning tree configuration program as set forth in claim 55, wherein said table comprises:

a destination MAC address field which describes the destination MAC address; and

an output port field which describes an output destination port corresponding to the destination MAC address.

57. (Allowable) The spanning tree configuration program as set forth in claim 58, comprising:

the transfer function of determining an output destination port in every partial network, based on an identification tag of an input frame;

the multiphase tree manager function of configuring a spanning tree for every partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a function of connecting said multiphase tree manager function and said transfer function through a virtual port for putting the output port toward the self-partial network together.

58. (Rejected) The spanning tree configuration program as set forth in claim 48, comprising a failure detecting function of detecting a failure through transmission and receipt of keep alive frames.

59. (Rejected) The spanning tree configuration program as set forth in claim 58, wherein said failure detecting function comprises:

a signal separating function of separating the keep alive frames from the other frame; and

a keep alive signal transmitting/receiving function of transmitting and receiving the keep alive frames.

60. (Rejected) The spanning tree configuration program as set forth in claim 58, comprising a blocking function of cutting off the port at a time of double failure.

61. (Rejected) The spanning tree configuration program as set forth in claim 58, comprising:

a transfer function of determining an output destination port in every said partial network, based on an identification tag of the input frame;

a multiphase tree manager function of configuring a spanning tree for every said partial network, according to the spanning tree protocol in every identification tag of the input frame; and

a tag operating function of inserting and deleting an identification tag.

62. (Canceled)

63. (Allowable) The network system as set forth in claim 16, wherein, when transmitting a control signal (control signals) of said spanning tree protocol to a node adjacent to the self-node and connected to both said partial network and said other adjacent network,

the coherent MAC address of the above node is transmitted as the destination of the control signal (control signals) of said spanning tree protocol.